Remarks

Applicants respectfully request reconsideration of the present application in view of the foregoing amendments and the following remarks. Claims 1, 3-13, 15-24, 26, 28, 31-36, 38, and 39 are pending in the application. Claims 1, 3-13, 15-24, 26, 28, 31-36, 38, and 39 are rejected. No claims have been allowed. Claims 1, 20, 31, and 36 are independent. Claims 1, 20, 31, 36, 38, and 39 have been amended.

Request for Review of Information Disclosure Statement

Applicants note that the Action does not include an initialed copy of the Form 1449 which accompanied the Information Disclosure Statement filed on May 3, 2006. Applicants respectfully request that the Examiner provide initialed copies of the indicated Forms 1449.

Cited Art

The Action cites

1. Gordon et al, U.S. Patent No. 6,560,774 (hereinafter "Gordon");
2. "Inside Microsoft®.NET IL Assembler," (hereinafter "Microsoft IL"); and
3. Syme et al., U.S. Patent No. 7,346,901 (hereinafter "Syme").

Claim Objections

Claims 38 and 39 are objected to for minor informalities. Claims 38 and 39 have been amended and should now be allowable. Applicants respectfully request withdrawal of the claim objections.

Claim Rejections under 35 U.S.C. § 103(a)

The Action rejects claims 1, 3-13, 15-24, 26, 28, and 31-35 under 35 U.S.C § 103(a) as unpatentable over Gordon in view of Microsoft IL.

The Action rejects claims 36, 38, and 39 under 35 U.S.C § 103(a) as unpatentable over Microsoft IL in view of Syme.

Claims 1, 3-13, and 15-19 are Allowable Over Gordon in View of Microsoft IL.

Claims 1, 3-13, and 15-19 are rejected as unpatentable over Gordon in view of Microsoft IL. Applicants respectfully disagree and traverse the rejections.

Claim 1 recites a method of representing type information for a typed intermediate language via objects of classes in a class hierarchy, wherein the class hierarchy comprises at least one class and a plurality of sub-classes for representing different type classifications, the method comprising, in part:

> instantiating one or more objects of one or more of the sub-classes of the hierarchy, wherein the one or more sub-classes represent classifications of types for the typed intermediate language;
>
> . . .
>
> wherein the classifications of types comprises a primitive type associated with a primitive type size, and wherein the primitive type size is settable to represent a constant size, the primitive type size is settable to represent a symbolic size, and the primitive type size is settable to represent an unknown size.

As the Examiner notes, "Gordon does not explicitly disclose the classifications of types comprises a primitive type associated with a size, and wherein the size is settable to represent a constant size, the size is settable to represent a symbolic size, and the size is settable to represent an unknown size." (Action, page 4.) Applicants agree. However, the Examiner alleges that Microsoft IL cures these deficiencies in Gordon. Applicants respectfully disagree. Specifically, Microsoft IL does not teach or suggest "wherein the primitive type size is settable to represent a symbolic size, and the primitive type size is settable to represent an unknown size" as recited by amended claim 1.

The action alleges that chapter 7, pages 6 and 7 of Microsoft IL describe the symbolic size and the unknown size, respectively. However, an array (vector or multidimensional array) is not a primitive type in MSIL. Instead, Microsoft IL states:

> The common language runtime recognizes two types of arrays: vectors and multidimensional arrays, as described in Table 7-3. Vectors are single-dimensional arrays with a zero lower bound. Multidimensional arrays, which I'll refer to as *arrays*, can have more than one dimension and nonzero lower bounds. **Neither of these two types of arrays has a respective .NET Framework type associated.**
> (Microsoft IL, chapter 7, page 5. Emphasis added by Applicants.)

As Applicants understand Microsoft IL, arrays are higher-level types, and their elements are of one of the primitive types (any primitive except a "void" primitive type). Microsoft IL states: "Vector encoding is very simple. E_T_SZARRAY followed by the encoding of the underlying type, which can be anything except void." (Microsoft IL, chapter 7, page 5.)

The lower and upper bounds described in Microsoft IL are used when instantiating a multidimensional array, but they are not a primitive type size because arrays are not primitive types in Microsoft IL.

Therefore, even if for the sake of argument the arrays are associated with sizes settable to represent a symbolic size or an unknown size, Microsoft IL does not and can not teach or suggest wherein "**the primitive type size** is settable to represent a symbolic size, and wherein **the primitive type size** is settable to represent an unknown size" as recited by amended claim 1 because Microsoft IL is referring to arrays which are explicitly stated to NOT have a .NET Framework type associated with the arrays.

Because Gordon and Microsoft IL, whether considered separately or in combination with each other, do not teach or suggest the above recited language of claim 1, claim 1 is allowable over Gordon and Microsoft IL. Dependent claims 3-13 and 15-19 are allowable at least because they depend from claim 1. Applicants respectfully request withdrawal of the § 103 rejections and allowance of claims 1, 3-13, and 15-19.

Claims 20-24, 26, and 28 are Allowable Over Gordon in View of Microsoft IL.

Claims 20-24, 26, and 28 are rejected as unpatentable over Gordon in view of Microsoft IL. Applicants respectfully disagree and traverse the rejections.

Amended independent claim 20 recites a computer-readable medium having a software program thereon, the program comprising computer executable instructions for implementing a method for representing type information for a typed intermediate language using a class hierarchy for representing different type classifications, the method comprising, in part:

> defining a programming class of the class hierarchy as 'PrimType', wherein the programming class represents primitive type information for the typed intermediate language;
>
> associating a size with instances of the 'PrimType' class, wherein the size is settable to represent an actual size of instances of the 'PrimType' class, settable to represent a symbolic size of instances of the 'PrimType' class, and settable to represent an unknown size of instances of the 'PrimType' class.

Neither Gordon nor Microsoft IL teach or suggest the above recited language of independent claim 20. Specifically neither reference describes associating a size with instances of the 'PrimType' class, wherein the size is "settable to represent a symbolic size of instances of the 'PrimType' class, and settable to represent an unknown size of instances of the 'PrimType' class."

As the Examiner notes, Gordon does not explicitly disclose primitive type sizes settable to represent a constant size, a symbolic size, and an unknown size. (Action, page 4.) Applicants agree. However, the Examiner alleges that Microsoft IL cures these deficiencies in Gordon. Applicants respectfully disagree.

As described above, Applicants understand Microsoft IL to describe arrays that are explicitly stated to not be associated with a .NET Framework type. Therefore, even if for the sake of argument the arrays are associated with sizes settable to represent a symbolic size or an unknown size, Microsoft IL does not and can not teach or suggest primitive type information wherein a size is associated with instances of the 'PrimType' class, which represents primitive type info, and the size is "settable to represent a symbolic size of instances of the 'PrimType' class, and settable to represent an unknown size of instances of the 'PrimType' class" as recited by amended independent claim 20 because Microsoft IL is referring to arrays which are explicitly stated to NOT have a .NET Framework type associated with the arrays.

Because Gordon and Microsoft IL, whether considered separately or in combination with each other, do not teach or suggest the above recited language of independent claim 20, claim 20 is allowable over Gordon and Microsoft IL. Dependent claims 21-24, 26, and 28 are allowable at least because they depend from claim 20. Applicants respectfully request withdrawal of the § 103 rejections and allowance of claims 20-24, 26, and 28.

Claims 31-35 are Allowable Over Gordon in View of Microsoft IL.

Claims 31-35 are rejected as unpatentable over Gordon in view of Microsoft IL. Applicants respectfully disagree and traverse the rejections.

Amended independent claim 31 recites a method for representing type information for a typed intermediate language using a class hierarchy by programmatically defining a type representation, the method comprising, in part:

> defining a plurality of classes hierarchically below the base class, wherein the plurality of classes represent type information for the typed intermediate language, and wherein the plurality of classes represent at least pointer types, container types and function types of a plurality of programming languages, and wherein the plurality of classes further comprise primitive types and the primitive types are associated with a primitive type size settable to represent a constant size, settable to represent a symbolic size, and settable to represent an unknown size.

Neither Gordon nor Microsoft IL teach or suggest the above recited language of independent claim 31. Specifically neither reference describes "a primitive type size . . . settable to represent a symbolic size, and settable to represent an unknown size."

As the Examiner notes, Gordon does not explicitly disclose primitive type sizes settable to represent a constant size, a symbolic size, and an unknown size. (Action, page 4.) Applicants agree. However, the Examiner alleges that Microsoft IL cures these deficiencies in Gordon. Applicants respectfully disagree.

As described above, Applicants understand Microsoft IL to describe arrays that are explicitly stated to not be associated with a .NET Framework type. Therefore, even if for the sake of argument the arrays are associated with sizes settable to represent a symbolic size or an unknown size, Microsoft IL does not and can not teach or suggest "a primitive type size . . . settable to represent a symbolic size, and settable to represent an unknown size" as recited by amended independent claim 31 because Microsoft IL is referring to arrays which are explicitly stated to NOT have a .NET Framework type associated with the arrays.

Because Gordon and Microsoft IL, whether considered separately or in combination with each other, do not teach or suggest the above recited language of independent claim 31, claim 31 is allowable over Gordon and Microsoft IL. Dependent claims 31-35 are allowable at least because they depend from claim 31. Applicants respectfully request withdrawal of the § 103 rejections and allowance of claims 31-35.

Claims 36, 38, and 39 are Allowable Over Microsoft IL in view of Syme.

Claims 36, 38, and 39 are rejected as unpatentable over Microsoft IL in view of Syme. Applicants respectfully disagree and traverse the rejections.

Independent claim 36 recites a computer-readable medium having a software program thereon, the program comprising computer executable instructions for implementing a method for representing type information for a typed intermediate language using a class hierarchy for representing different type classifications, the method comprising, in part:

> defining a programming class of the class hierarchy as 'PrimType', wherein an object of class 'PrimType' is a type representation for the typed intermediate language for primitive types in a section of code written in one of a plurality of programming languages;
>
> . . .;
>
> wherein the object of class 'PrimType' is associated with a size settable to represent a constant size for the object of class 'PrimType', settable to represent a symbolic size for the object of class 'PrimType', and settable to represent an unknown size for the object of class 'PrimType'.

Neither Microsoft IL nor Syme teach or suggest the above recited language of independent claim 36. Specifically neither reference describes an object of class 'PrimType' which is a type representation for the typed intermediate language for primitive types, and wherein the object of class 'PrimType' is associated with a size "settable to represent a symbolic size for the object of class 'PrimType', and settable to represent an unknown size for the object of class 'PrimType'" as recited in independent claim 36.

As described above, Applicants understand Microsoft IL to describe arrays that are explicitly stated to not be associated with a .NET Framework type. Therefore, even if for the sake of argument the arrays are associated with sizes settable to represent a symbolic size or an unknown size, Microsoft IL does not and can not teach or suggest "an object of class 'PrimType' [which] is a type representation for the typed intermediate language for primitive types . . . wherein the object of class 'PrimType' is associated with a size . . . settable to represent a symbolic size for the object of class 'PrimType', and settable to represent an unknown size for the object of class 'PrimType'" as recited by amended independent claim 36 because Microsoft IL is referring to arrays which are explicitly stated to NOT have a .NET Framework type associated with the arrays.

Syme does not cure this deficiency of Microsoft IL, nor is it cited by the Examiner for that purpose.

Because Microsoft IL and Syme, whether considered separately or in combination with each other, do not teach or suggest each and every element of independent claim 36, claim 36 is allowable. Dependent claims 38 and 39 are allowable at least because they depend from independent claim 36.

Applicants respectfully request withdrawal of the § 103 rejections and allowance of claims 36, 38, and 39.

Interview Request

If the claims are not found by the Examiner to be allowable, the Examiner is requested to call the undersigned attorney to set up an interview to discuss this application.

Conclusion

The claims in their present form should be allowable. Such action is respectfully requested.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Telephone: (503) 595-5300
Facsimile: (503) 595-5301

By ______________________
Cory A. Jones
Registration No. 55,307